

Human happiness analysis (based on ESS data)

Sebastian Boruch

06.04.2018

Introduction

This report presents analysis of data from the European Social Survey (ESS) in terms of the impact of people from respondent's environment on his/her happiness level. For more details on ESS data check: <http://www.europeansocialsurvey.org/>

ESS data analysis

Variables were selected in such a way that they were able to describe the respondent's social situation - the number of people close to him in his environment and his approach to people. The collection also included variables informing about the country of origin of parents. It is easy to say that people whose parents do not come from the current country of residence of the respondent may have greater difficulties in social interactions due to possible language barriers and cultural differences. It was recognized that these factors also influence the social status of the respondent.

```
ess <- read.dta("ESS8e01.dta")
```

```
myvars <- c("chldhhe", "dvrcdeva", "facntr", "happy", "hhmb", "inprdsc",  
"iphlppl", "iplylfr", "iprspot", "ipudrst", "jbspv", "lvgptnea", "mocntr",  
"rshpsts", "sclmeet")
```

```
mydata <- ess[myvars]
```

```
attach(mydata)
```

```
summary(mydata)
```

##	chldhhe		dvrcdeva		facntr	
##	Yes	:11671	Yes	: 5936	Yes	:28327
##	No	:10907	No	:28695	No	: 6291
##	Not applicable:	12086	Refusal	: 149	Refusal	: 22
##	Refusal	: 93	Don't know:	9	Don't know:	148
##	Don't know	: 14	No answer :	8	No answer :	9
##	No answer	: 26	NA's	: 40	NA's	: 40
##	NA's	: 40				
##	happy		hhmb		inprdsc	
##	8	:10297	Min.	: 1.000	4-6	:9555
##	9	: 6612	1st Qu.:	2.000	3	:8327
##	7	: 6194	Median :	2.000	2	:6748
##	Extremely happy:	4076	Mean	: 2.769	1	:4841

```

## 5 : 2673 3rd Qu.: 4.000 7-9 :1976
## (Other) : 4945 Max. :99.000 (Other):3350
## NA's : 40 NA's :40 NA's : 40
## iphlppl iplylfr
## Like me :14659 Like me :15048
## Very much like me: 8983 Very much like me:12478
## Somewhat like me : 7192 Somewhat like me : 4646
## A little like me : 2583 A little like me : 1546
## Not like me : 720 Not like me : 459
## (Other) : 660 (Other) : 620
## NA's : 40 NA's : 40
## iprspot ipudrst jbspv
## Like me :8544 Like me :15035 Yes : 9906
## Somewhat like me :8341 Somewhat like me : 7534 No :22512
## A little like me :6330 Very much like me: 7103 Not applicable: 2255
## Not like me :6190 A little like me : 3125 Refusal : 75
## Very much like me:3329 Not like me : 1209 Don't know : 35
## (Other) :2063 (Other) : 791 No answer : 14
## NA's : 40 NA's : 40 NA's : 40
## lvgptnea mocntr
## Yes :10610 Yes :28701
## No :20047 No : 6027
## Not applicable: 3904 Refusal : 17
## Refusal : 197 Don't know: 49
## Don't know : 21 No answer : 3
## No answer : 18 NA's : 40
## NA's : 40
## rshpsts
## Legally married :16229
## Not applicable :14277
## Living with my partner - not legally recognised: 3253
## Living with my partner - legally recognised : 651
## In a legally registered civil union : 225
## (Other) : 162
## NA's : 40
## sclmeet
## Several times a week :9881
## Several times a month:7186
## Once a week :6536
## Every day :4723
## Once a month :3427
## (Other) :3044
## NA's : 40

```

Recoding variables

The data was downloaded in .dta format and read using the "foreign" package. Except HHMMB variable, each of the variables had a "factor variable" format, so the program did not read them as numeric variables, and as variables with categories without a specific

order. In order to create a regression model, it was decided to recode each of the variables except HHMMB. The process was as follows:

```
mydata$hhmb[mydata$hhmb > 20] <- NA

happy2 <- as.factor(happy)
happy2 <- revalue(happy2, c("Extremely unhappy"=0))
happy2 <- revalue(happy2, c("Extremely happy"=10))
happy2 <- revalue(happy2, c("Refusal"=NA))
happy2 <- revalue(happy2, c("Don't know"=NA))
happy2 <- revalue(happy2, c("No answer"=NA))
happy2 <- as.numeric(levels(happy2))[happy2]

chldhhe2 <- as.factor(chldhhe)
chldhhe2 <- revalue(chldhhe2, c("Yes"=1))
chldhhe2 <- revalue(chldhhe2, c("No"=0))
chldhhe2 <- revalue(chldhhe2, c("Not applicable"=NA))
chldhhe2 <- revalue(chldhhe2, c("Refusal"=NA))
chldhhe2 <- revalue(chldhhe2, c("Don't know"=NA))
chldhhe2 <- revalue(chldhhe2, c("No answer"=NA))
chldhhe2 <- as.numeric(levels(chldhhe2))[chldhhe2]

jbspv2 <- as.factor(jbspv)
jbspv2 <- revalue(jbspv2, c("Yes"=1))
jbspv2 <- revalue(jbspv2, c("No"=0))
jbspv2 <- revalue(jbspv2, c("Not applicable"=NA))
jbspv2 <- revalue(jbspv2, c("Refusal"=NA))
jbspv2 <- revalue(jbspv2, c("Don't know"=NA))
jbspv2 <- revalue(jbspv2, c("No answer"=NA))
jbspv2 <- as.numeric(levels(jbspv2))[jbspv2]

lvgtptnea2 <- as.factor(lvgtptnea)
lvgtptnea2 <- revalue(lvgtptnea2, c("Yes"=1))
lvgtptnea2 <- revalue(lvgtptnea2, c("No"=0))
lvgtptnea2 <- revalue(lvgtptnea2, c("Not applicable"=NA))
lvgtptnea2 <- revalue(lvgtptnea2, c("Refusal"=NA))
lvgtptnea2 <- revalue(lvgtptnea2, c("Don't know"=NA))
lvgtptnea2 <- revalue(lvgtptnea2, c("No answer"=NA))
lvgtptnea2 <- as.numeric(levels(lvgtptnea2))[lvgtptnea2]

dvrcdeva2 <- as.factor(dvrcdeva)
dvrcdeva2 <- revalue(dvrcdeva2, c("Yes"=1))
dvrcdeva2 <- revalue(dvrcdeva2, c("No"=0))
dvrcdeva2 <- revalue(dvrcdeva2, c("Refusal"=NA))
dvrcdeva2 <- revalue(dvrcdeva2, c("Don't know"=NA))
dvrcdeva2 <- revalue(dvrcdeva2, c("No answer"=NA))
dvrcdeva2 <- as.numeric(levels(dvrcdeva2))[dvrcdeva2]
```

```

facntr2 <- as.factor(facntr)
facntr2 <- revalue(facntr2, c("Yes"=1))
facntr2 <- revalue(facntr2, c("No"=0))
facntr2 <- revalue(facntr2, c("Refusal"=NA))
facntr2 <- revalue(facntr2, c("Don't know"=NA))
facntr2 <- revalue(facntr2, c("No answer"=NA))
facntr2 <- as.numeric(levels(facntr2))[facntr2]

mocntr2 <- as.factor(mocntr)
mocntr2 <- revalue(mocntr2, c("Yes"=1))
mocntr2 <- revalue(mocntr2, c("No"=0))
mocntr2 <- revalue(mocntr2, c("Refusal"=NA))
mocntr2 <- revalue(mocntr2, c("Don't know"=NA))
mocntr2 <- revalue(mocntr2, c("No answer"=NA))
mocntr2 <- as.numeric(levels(mocntr2))[mocntr2]

inprdsc2 <- as.factor(inprdsc)
inprdsc2 <- revalue(inprdsc2, c("None"=0))
inprdsc2 <- revalue(inprdsc2, c("4-6"=5))
inprdsc2 <- revalue(inprdsc2, c("7-9"=8))
inprdsc2 <- revalue(inprdsc2, c("10 or more"=10))
inprdsc2 <- revalue(inprdsc2, c("Refusal"=NA))
inprdsc2 <- revalue(inprdsc2, c("Don't know"=NA))
inprdsc2 <- revalue(inprdsc2, c("No answer"=NA))
inprdsc2 <- as.numeric(levels(inprdsc2))[inprdsc2]

iphlppl2 <- as.factor(iphlppl)
iphlppl2 <- revalue(iphlppl2, c("Very much like me"=5))
iphlppl2 <- revalue(iphlppl2, c("Like me"=4))
iphlppl2 <- revalue(iphlppl2, c("Somewhat like me"=3))
iphlppl2 <- revalue(iphlppl2, c("A little like me"=2))
iphlppl2 <- revalue(iphlppl2, c("Not like me"=1))
iphlppl2 <- revalue(iphlppl2, c("Not like me at all"=0))
iphlppl2 <- revalue(iphlppl2, c("Refusal"=NA))
iphlppl2 <- revalue(iphlppl2, c("Don't know"=NA))
iphlppl2 <- revalue(iphlppl2, c("No answer"=NA))
iphlppl2 <- as.numeric(levels(iphlppl2))[iphlppl2]

iplylfr2 <- as.factor(iplylfr)
iplylfr2 <- revalue(iplylfr2, c("Very much like me"=5))
iplylfr2 <- revalue(iplylfr2, c("Like me"=4))
iplylfr2 <- revalue(iplylfr2, c("Somewhat like me"=3))
iplylfr2 <- revalue(iplylfr2, c("A little like me"=2))
iplylfr2 <- revalue(iplylfr2, c("Not like me"=1))
iplylfr2 <- revalue(iplylfr2, c("Not like me at all"=0))
iplylfr2 <- revalue(iplylfr2, c("Refusal"=NA))
iplylfr2 <- revalue(iplylfr2, c("Don't know"=NA))
iplylfr2 <- revalue(iplylfr2, c("No answer"=NA))
iplylfr2 <- as.numeric(levels(iplylfr2))[iplylfr2]

```

```

iprspot2 <- as.factor(iprspot)
iprspot2 <- revalue(iprspot2, c("Very much like me"=5))
iprspot2 <- revalue(iprspot2, c("Like me"=4))
iprspot2 <- revalue(iprspot2, c("Somewhat like me"=3))
iprspot2 <- revalue(iprspot2, c("A little like me"=2))
iprspot2 <- revalue(iprspot2, c("Not like me"=1))
iprspot2 <- revalue(iprspot2, c("Not like me at all"=0))
iprspot2 <- revalue(iprspot2, c("Refusal"=NA))
iprspot2 <- revalue(iprspot2, c("Don't know"=NA))
iprspot2 <- revalue(iprspot2, c("No answer"=NA))
iprspot2 <- as.numeric(levels(iprspot2))[iprspot2]

ipudrst2 <- as.factor(ipudrst)
ipudrst2 <- revalue(ipudrst2, c("Very much like me"=5))
ipudrst2 <- revalue(ipudrst2, c("Like me"=4))
ipudrst2 <- revalue(ipudrst2, c("Somewhat like me"=3))
ipudrst2 <- revalue(ipudrst2, c("A little like me"=2))
ipudrst2 <- revalue(ipudrst2, c("Not like me"=1))
ipudrst2 <- revalue(ipudrst2, c("Not like me at all"=0))
ipudrst2 <- revalue(ipudrst2, c("Refusal"=NA))
ipudrst2 <- revalue(ipudrst2, c("Don't know"=NA))
ipudrst2 <- revalue(ipudrst2, c("No answer"=NA))
ipudrst2 <- as.numeric(levels(ipudrst2))[ipudrst2]

sclmeet2 <- as.factor(sclmeet)
sclmeet2 <- revalue(sclmeet2, c("Never"=0))
sclmeet2 <- revalue(sclmeet2, c("Less than once a month"=1))
sclmeet2 <- revalue(sclmeet2, c("Once a month"=2))
sclmeet2 <- revalue(sclmeet2, c("Several times a month"=3))
sclmeet2 <- revalue(sclmeet2, c("Once a week"=4))
sclmeet2 <- revalue(sclmeet2, c("Several times a week"=5))
sclmeet2 <- revalue(sclmeet2, c("Every day"=6))
sclmeet2 <- revalue(sclmeet2, c("Refusal"=NA))
sclmeet2 <- revalue(sclmeet2, c("Don't know"=NA))
sclmeet2 <- revalue(sclmeet2, c("No answer"=NA))
sclmeet2 <- as.numeric(levels(sclmeet2))[sclmeet2]

mydata2 <- cbind.data.frame(chldhhe2, dvrdeva2, facntr2, happy2, inprdsc2,
  iphlpl2, iplylfr2, iprspot2, ipudrst2, jbspv2, lvgptnea2, mocntr2, sclmeet2,
  hhmmb)
mydata2 <- rename(mydata2, c("chldhhe2"="chldhhe", "dvrdeva2"="dvrdeva",
  "facntr2"="facntr", "happy2"="happy", "inprdsc2"="inprdsc",
  "iphlpl2"="iphlpl", "iplylfr2"="iplylfr", "iprspot2"="iprspot",
  "ipudrst2"="ipudrst", "jbspv2"="jbspv", "lvgptnea2"="lvgptnea",
  "mocntr2"="mocntr", "sclmeet2"="sclmeet"))
# mydata2- only numerical variables

```

In addition, for each response variable: "I don't know", the refusal or lack of answer were marked as "NA". The average percentage of NA's for one variable was:

```
mean(is.na(mydata2))
```

```
## [1] 0.0460594
```

Replacing missing values

Therefore, it was decided to replace the gaps with the values calculated by PMM. Predictive Mean Matching (PMM) is a semiparametric data imputation method. It is similar to the regression method, with the difference that for each missing value it randomly assigns a value from the set of observed values whose predicted values are the closest to the predicted value for the missing value from the simulated regression model. The PMM method ensures that the imputed values are reliable; it may be more appropriate than the regression method if the assumption of a normal distribution is violated.

```
imputed_Data <- mice(mydata2, m=5, maxit = 20, method = 'pmm') # method
predictive mean matching; Bodner (2008) and White et al (2011) %NA=imputation
number(m)- here: 5
```

```
summary(imputed_Data)
```

```
## Multiply imputed data set
```

```
## Call:
```

```
## mice(data = mydata2, m = 5, method = "pmm", maxit = 20)
```

```
## Number of multiple imputations: 5
```

```
## Missing cells per column:
```

##	chldhhe	dvrdeva	facntr	happy	inprdsc	iphlppl	iplylfr	iprspot
##	12259	206	219	189	340	538	540	685
##	ipudrst	jbspv	lvgtptnea	mocntr	sclmeet	hhmmb		
##	573	2419	4180	109	167	40		

```
## Imputation methods:
```

##	chldhhe	dvrdeva	facntr	happy	inprdsc	iphlppl	iplylfr	iprspot
##	"pmm"	"pmm"	"pmm"	"pmm"	"pmm"	"pmm"	"pmm"	"pmm"
##	ipudrst	jbspv	lvgtptnea	mocntr	sclmeet	hhmmb		
##	"pmm"	"pmm"	"pmm"	"pmm"	"pmm"	"pmm"		

```
## VisitSequence:
```

##	chldhhe	dvrdeva	facntr	happy	inprdsc	iphlppl	iplylfr	iprspot
##	1	2	3	4	5	6	7	8
##	ipudrst	jbspv	lvgtptnea	mocntr	sclmeet	hhmmb		
##	9	10	11	12	13	14		

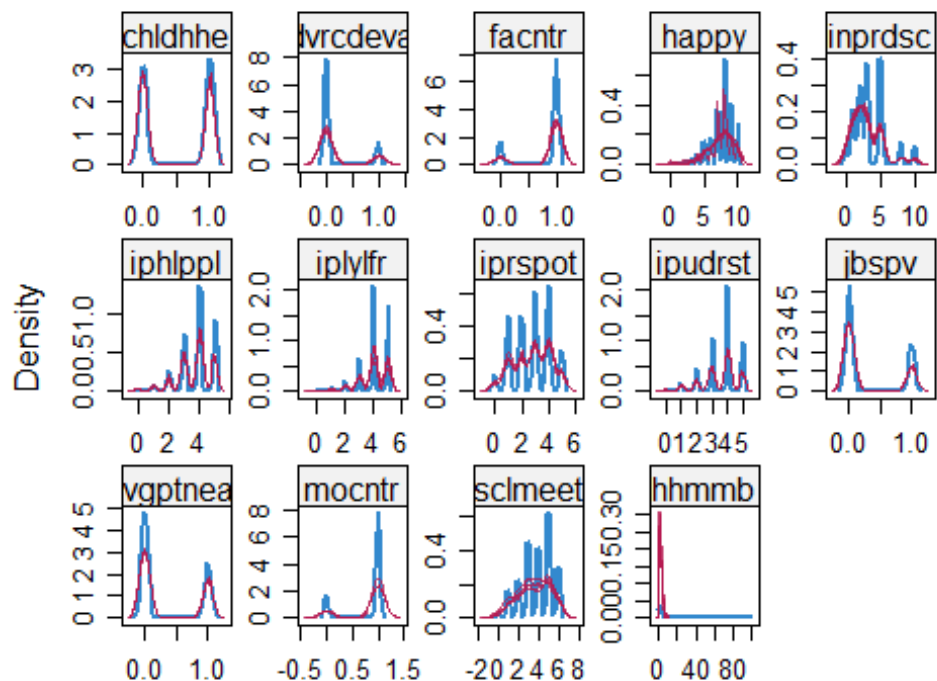
```
## PredictorMatrix:
```

##		chldhhe	dvrdeva	facntr	happy	inprdsc	iphlppl	iplylfr	iprspot
##	chldhhe	0	1	1	1	1	1	1	1
##	dvrdeva	1	0	1	1	1	1	1	1
##	facntr	1	1	0	1	1	1	1	1
##	happy	1	1	1	0	1	1	1	1
##	inprdsc	1	1	1	1	0	1	1	1
##	iphlppl	1	1	1	1	1	0	1	1
##	iplylfr	1	1	1	1	1	1	0	1
##	iprspot	1	1	1	1	1	1	1	0
##	ipudrst	1	1	1	1	1	1	1	1

```
## jbspv      1      1      1      1      1      1      1      1
## lvgptnea   1      1      1      1      1      1      1      1
## mocntr     1      1      1      1      1      1      1      1
## sclmeet    1      1      1      1      1      1      1      1
## hhmmb      1      1      1      1      1      1      1      1
##           ipudrst jbspv lvgptnea mocntr sclmeet hhmmb
## chldhhe    1      1      1      1      1      1
## dvrdeva    1      1      1      1      1      1
## facntr     1      1      1      1      1      1
## happy      1      1      1      1      1      1
## inprdsc    1      1      1      1      1      1
## iphlpl     1      1      1      1      1      1
## iplylfr    1      1      1      1      1      1
## iprspot    1      1      1      1      1      1
## ipudrst    0      1      1      1      1      1
## jbspv      1      0      1      1      1      1
## lvgptnea    1      1      0      1      1      1
## mocntr     1      1      1      0      1      1
## sclmeet    1      1      1      1      0      1
## hhmmb      1      1      1      1      1      0
## Random generator seed value: NA
```

```
densityplot(imputed_Data)
```

In the below chart, the results of each of the imputed sets were marked with maroon color. As one can see, each of the datasets is very similar to each other, which means that one can choose any of the new datasets received. It was decided to choose No. 2.

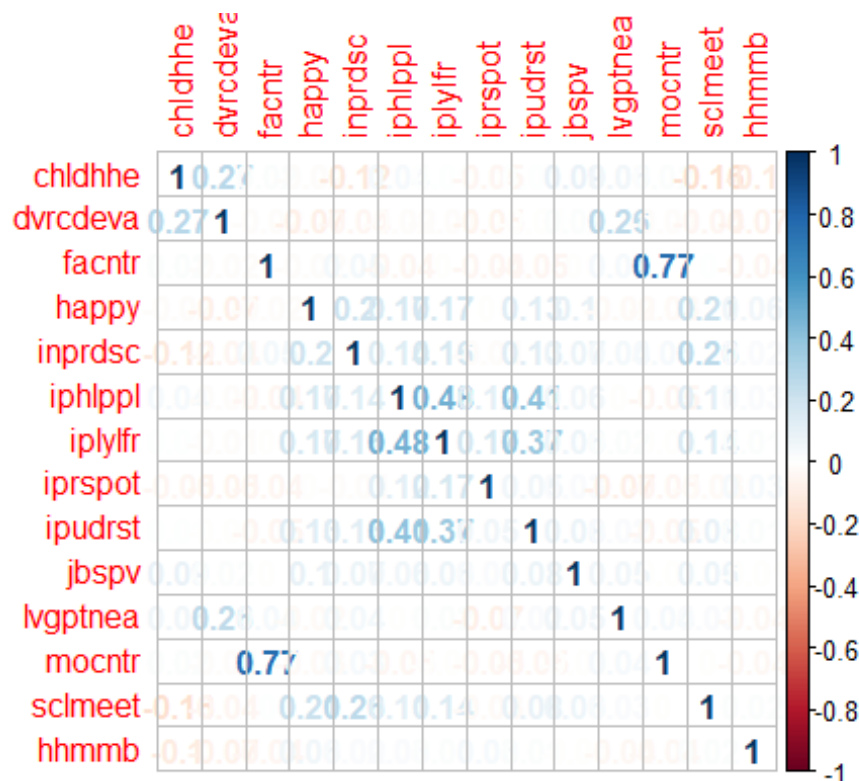


```
completedata <- complete(imputed_Data,2)
detach(mydata)
attach(mydata2)
```

Collinearity

After processing the set and solving the problem of data gaps, the collinearity analysis of variables was started. For this purpose, a correlation diagram of each of the variables was made using the `corrplot()` function from the `corr` package. The graph shows the values of Pearson coefficients:

```
# collinearity
newdatacor = cor(completedata[1:14], method= "pearson")
corrplot(newdatacor, method = "number")
```



The above graph shows a strong correlation between MOCNTR and FACNTR variables, that is the mother and father's country of origin. It was decided to take into account the interaction between these variables by creating a new variable $PARNTR = MOCNTR * FACNTR$

```
parntr<- mocntr*facntr
completedata <- mutate(completedata, parntr= mocntr*facntr)

corrvars <- names(completedata) %in% c("mocntr", "facntr")
regressiondata <- completedata[!corrvars]
```


Linear regression

The results of linear regression with the inclusion of each of the explanatory variables were as follows:

```
# Linear regression
lm.happy <- lm(happy~., data=regressiondata)
summary(lm.happy)

##
## Call:
## lm(formula = happy ~ ., data = regressiondata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8.6174 -0.8196  0.2261  1.1374  4.7206
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.145131   0.057360  89.699 < 2e-16 ***
## chldhhe      0.129081   0.019594   6.588 4.53e-11 ***
## dvrcldeva    -0.300032   0.026159 -11.469 < 2e-16 ***
## inprdsc      0.098674   0.004118  23.964 < 2e-16 ***
## iphlpl       0.131803   0.010897  12.095 < 2e-16 ***
## iplylfr      0.162561   0.011786  13.793 < 2e-16 ***
## iprspot     -0.039246   0.006874  -5.709 1.14e-08 ***
## ipudrst      0.058844   0.009738   6.043 1.53e-09 ***
## jbspv        0.272131   0.020225  13.455 < 2e-16 ***
## lvgptnea    -0.097564   0.020026  -4.872 1.11e-06 ***
## sclmeet      0.186905   0.006440  29.021 < 2e-16 ***
## hhmmb        0.021528   0.002362   9.113 < 2e-16 ***
## parntr      -0.080274   0.022590  -3.554 0.000381 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.713 on 34824 degrees of freedom
## Multiple R-squared:  0.1048, Adjusted R-squared:  0.1045
## F-statistic: 339.8 on 12 and 34824 DF, p-value: < 2.2e-16

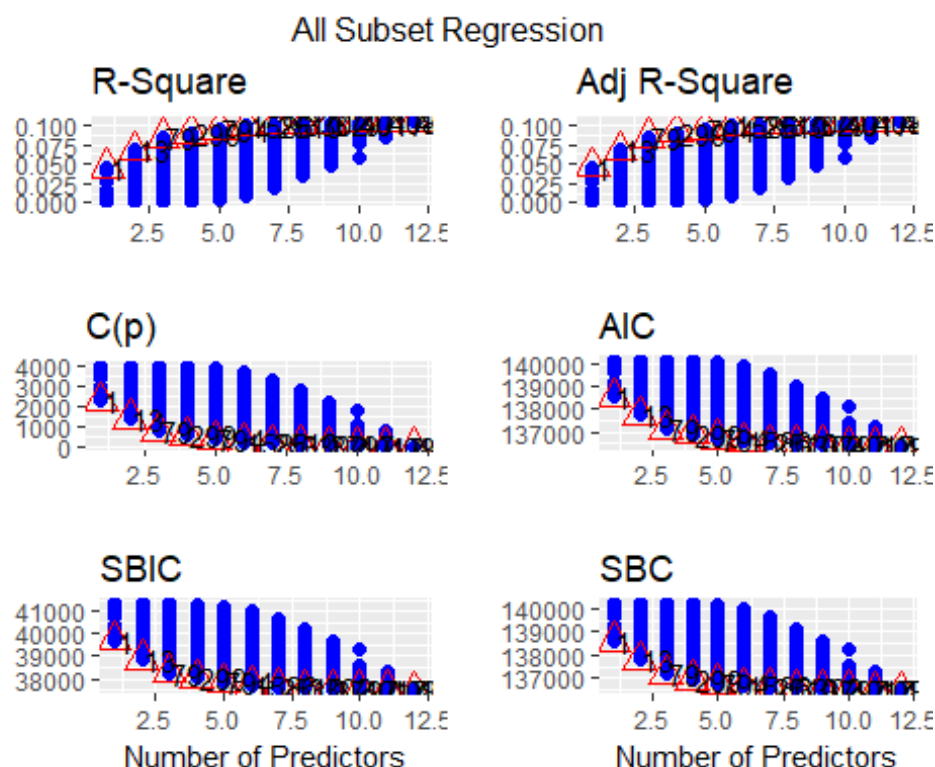
AIC(lm.happy)

## [1] 136372.8
```

Each of the variables turned out to be statistically significant at the confidence level of 95%. The R^2 value was only 12%. A small R^2 value does not mean that the model is not matched. A literal interpretation of this coefficient is as follows: the model explains the variance of the HAPPY variable in 12%. Therefore, one can deduct that the human environment influences happiness level of a person by 12%. It should be borne in mind that there are many factors influencing human happiness in a more direct way, e.g. income level, age, health or place of residence. The value of the F statistic is 390 and AIC-135844. These

statistics become useful when comparing with other models. Very small p-value means that at least one of the explanatory variables is statistically significant. As each variable turned out to be statistically significant, it is not necessary to create a model with a selection strategy.

```
k <- ols_all_subset(lm.happy)
plot(k)
```

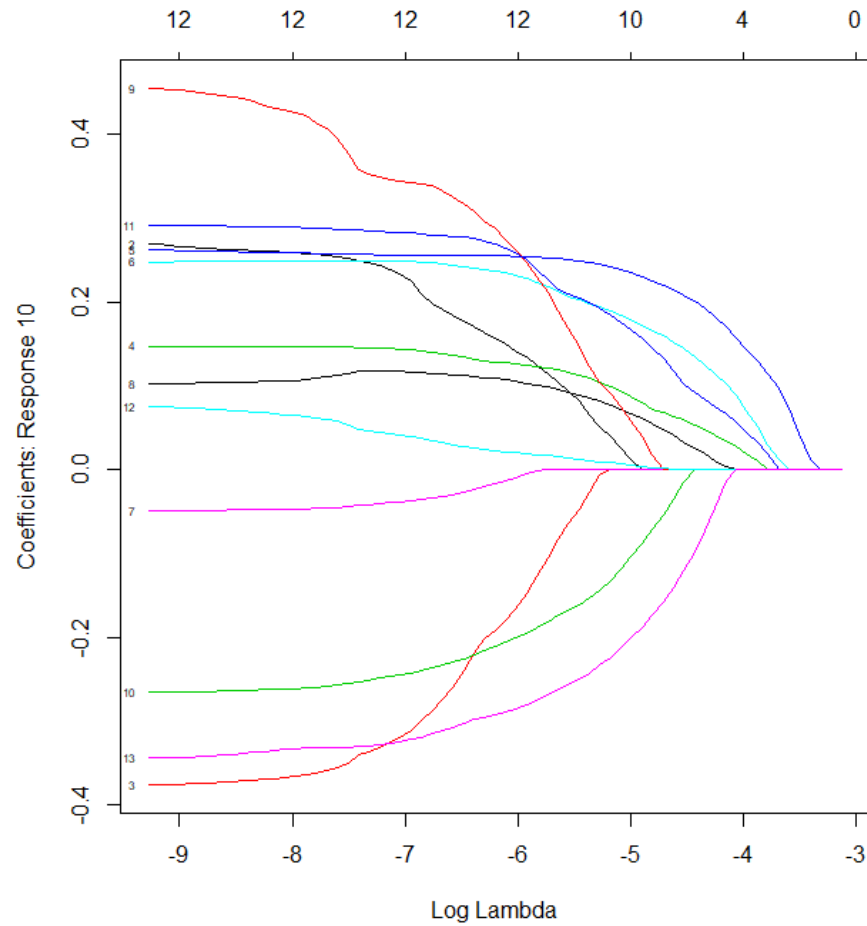


The above plot shows how the model was seeking for optimum.

ESS data analysis

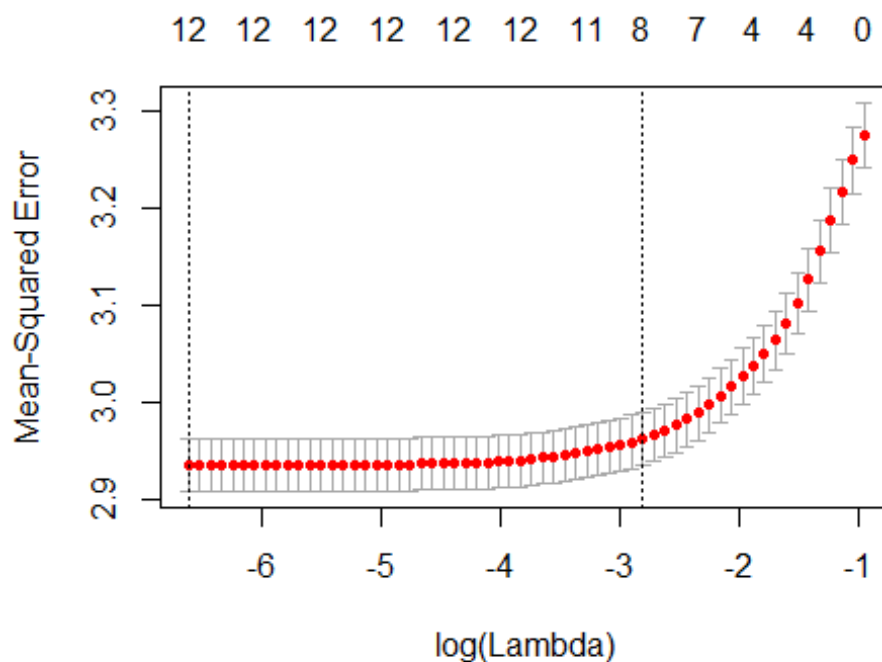
The second method used to detect dependences of explanatory variables with the variable HAPPY was LASSO regression. LASSO is a method of variable selection for statistical models. The LASSO method imposes a constraint on the sum of the absolute values of the model parameters, the sum must be less than the fixed value (upper limit).

```
# LASSO
x <- model.matrix(happy~., -1, data = regressiondata)
y <- regressiondata$happy
fit.lasso <- glmnet(x, y, family="multinomial")
plot(fit.lasso, xvar="lambda", label=TRUE)
```



Each curve represents the ratio in the model. The X axis is a lambda function, the penalty parameter. The y-axis gives the value of the coefficient. The graph shows how the coefficients "enter the model" (become non-zero) when the lambda changes. The graph shows that the most deviated or influential variables are variables 3 and 9, i.e. INPRDSC and LVGPTNEA, respectively.

```
cv.lasso <- cv.glmnet(x, y)
plot(cv.lasso)
```



```
coef(cv.lasso)

## 14 x 1 sparse Matrix of class "dgCMatrix"
##              1
## (Intercept)  5.542279819
## (Intercept)  .
## chldhhe      .
## dvrceva      -0.137458686
## inprdsc      0.081984429
## iphlppl      0.111959303
## iplylfr      0.127206143
## iprspot      .
## ipudrst      0.031807680
## jbspv        0.164841256
## lvgptnea     .
## sclmeet      0.157661214
## hhmb         0.006490678
## parntr       .
```

After using the LASSO method, the following explanatory variables remained in the model: DVRCDEVA, INPRDSC, IPHLPPL, IPLYLFR, IPUDRST, JBSPV, SCLMEET, HHMMB. Then another linear regression model was built using these variables. The results of the second linear regression model were as follows:

```
# Linear regression No. 2
lm.happy2 <- lm(happy~dvrcdeva+ inprdsc+ iphlppl+ iplylfr+ ipudrst+
                jbspv+ sclmeet+ hhmb, data=regressiondata)
summary(lm.happy2)

##
## Call:
## lm(formula = happy ~ dvrcdeva + inprdsc + iphlppl + iplylfr +
##     ipudrst + jbspv + sclmeet + hhmb, data = regressiondata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8.6258 -0.8206  0.2254  1.1398  4.8125
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.055155   0.050902  99.311 < 2e-16 ***
## dvrcdeva     -0.279314   0.024467 -11.416 < 2e-16 ***
## inprdsc       0.095772   0.004100  23.361 < 2e-16 ***
## iphlppl       0.135398   0.010876  12.450 < 2e-16 ***
## iplylfr       0.151995   0.011672  13.022 < 2e-16 ***
## ipudrst       0.060231   0.009740   6.184 6.32e-10 ***
## jbspv         0.278239   0.020132  13.821 < 2e-16 ***
## sclmeet       0.182158   0.006372  28.587 < 2e-16 ***
## hhmb          0.020373   0.002355   8.651 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.715 on 34828 degrees of freedom
## Multiple R-squared:  0.1019, Adjusted R-squared:  0.1017
## F-statistic: 494.2 on 8 and 34828 DF,  p-value: < 2.2e-16

AIC(lm.happy2)

## [1] 136476.8
```

As one can see, each variable is statistically significant, even at the 99% significance level. Unfortunately, R^2 has remained low and the result of the AIC-136476.7 criterion is almost the same as in the first model.

Conclusions

From the perspective of interpersonal interactions, the following variables influence the human level of happiness:

- free expression: 5.05. This means that with all other parameters equal to 0, the level of happiness would be 5.05 on a scale of 0-10,
- DVRCDEVA: -0.28. This means that those who divorce are 28% happier than those without divorce,

- INPRDSC: 0.096. A clear interpretation is complicated by the uneven scale of this variable. However, it can be stated that if the respondent has one more person for intimate confessions, his happiness level should increase by about 9%,
- IPHLPPL: 0.13. People who are 1/6 more empathetic, are 13% happier,
- IPLYLFR: 0.15. It can be deduced from this value that with the increase in human loyalty by 1/6, its happiness level increases by 13%,
- IPUDRST: 0.07. People for whom the understanding of other people is 1/6 more important- are 7% happier,
- JBSPV: 0.29. People who are responsible for managing other people are 29% happier,
- SCLMEET: 0.18. People who meet with family or friends 1/7 more often are 18% happier,
- HHMMB: 0.02. People who live with 1 more person in the household are 2% happier than those with 1 person less,

All the above variables are statistically significant, so it is reasonable to draw conclusions from their coefficients. However, the R^2 coefficient is only 10%, so the whole model explains the variance of the HAPPY variable only in 10%. Thus, it can be concluded that social interactions are an important factor affecting the well-being of a person, but certainly not the only one and not even the key one.