Shape my City: Political participation in Lucerne

Marlen Stalder

27.11.2020

Political participation can have a great influence on quality of life. Therefore I want to take a closer look at this subject. Do the demopgraphic factors have an influence on how good people know about possibilities for participation?

The analysis includes the following variables:

Survey questions

- Kennen Sie die Möglichkeiten, sich auf städtischer Ebene mit persönlichen Anliegen einzubringen
- Kennen Sie die Möglichkeiten, sich in städtische Projekte einzubringen?
- Kennen Sie die Möglichkeiten, städtische Projekte vorzuschlagen?

Demographics

- Geschlecht
- Altersgruppe
- Quartier
- Zuzug
- Bildung
- Haushalt
- Kinder_Schule

```
library(dplyr)
library(ggplot2)
library(gtable)
library(tree)
```

Load data

```
# load data
df <- read.csv("./cleaned_data.csv", sep=";", header=TRUE)
```

Prepare data

```r
#rename columns
df <- df %>% rename(knw.personal.matters = v_316,
                    sat.personal.matters = v_106,
                    knw.participate = v_212,
                    knw.propose = v_365,
                    sat.participate = v_366,
                    sat.propose = v_367,
                    Geschlecht = Sex
                    )

# create subset
df.small <- select(df, c("Geschlecht", "Altergruppe","Beziehungstatus","Quartier",
                         "Zuzug", "Bildung", "HH", "Kinder_Schule", "Erwerb",
                         "knw.personal.matters", "knw.participate", "knw.propose",
                         "sat.personal.matters", "sat.participate", "sat.propose"))

# have a look at values
df.small %>% count(knw.personal.matters)
df.small %>% count(knw.participate)
df.small %>% count(knw.propose)
df.small %>% count(sat.personal.matters)
df.small %>% count(sat.participate)
df.small %>% count(sat.propose)

df.small %>% count(Geschlecht)
df.small %>% count(Altergruppe)
df.small %>% count(Quartier)
df.small %>% count(Zuzug)
df.small %>% count(Bildung)
df.small %>% count(HH)
df.small %>% count(Kinder_Schule)
df.small %>% count(Erwerb)

# delete all rows without an answer
df.small <-  df.small[rowSums(df.small == -77 | df.small == 0)==0,,]

# convert all columns to factors
df.small <- replace(df.small, TRUE, lapply(df.small, factor))

# level names
Zuzug.levels = c("<2", "2-5", "6-10", "11-15", "16-20", "21-25", ">25")
Bildung.levels = c("Berufslehre", "Gymnasium", "Hochschule", "Andere")
Quartier.levels = c("Innenstadt", "Rechte Seeseite", "Linke Seeseite",
                    "Rechtes Reussufer", "Linkes Reussufer", "Littau")
Altergruppe.levels = c("0-9", "10-19", "20-29", "30-39", "40-49", "50-65", "65-84", "85+")
HH.levels = c("Einpersonenhaushalt", "Elternteil mit Kind(ern)", "Paar ohne Kinder",
              "Paar mit Kind(ern)", "anderer Mehrpersonenhaushalt")
```

Logistic Regression models

For the yes/no questions I appy a logicstic regression model.

```r
lm.knw.personal.matters <- glm(knw.personal.matters ~ Geschlecht + Altergruppe +
                                 Beziehungstatus + Quartier + Zuzug + Bildung +
                                 HH + Kinder_Schule + Erwerb,
                               family = "binomial",
                               data = df.small)

summary(lm.knw.personal.matters)
```

```
##
## Call:
## glm(formula = knw.personal.matters ~ Geschlecht + Altergruppe +
##     Beziehungstatus + Quartier + Zuzug + Bildung + HH + Kinder_Schule +
##     Erwerb, family = "binomial", data = df.small)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0102  -0.9699  -0.4034   0.9804   2.0843
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        0.90876    1.24061   0.733  0.46386
## Geschlecht2        0.51376    0.18735   2.742  0.00610 **
## Geschlecht3      -13.76713  882.74345  -0.016  0.98756
## Altergruppe2      -0.38530    1.14295  -0.337  0.73603
## Altergruppe3      -0.22794    1.13062  -0.202  0.84023
## Altergruppe4      -0.09337    1.12801  -0.083  0.93403
## Altergruppe5      -0.27004    1.13693  -0.238  0.81225
## Altergruppe6      -0.46686    1.14923  -0.406  0.68457
## Altergruppe7      -0.58405    1.19600  -0.488  0.62531
## Altergruppe8     -14.21551  882.74424  -0.016  0.98715
## Beziehungstatus2   0.11382    0.26848   0.424  0.67162
## Quartier2         -0.13453    0.29323  -0.459  0.64639
## Quartier3         -0.15382    0.26828  -0.573  0.56640
## Quartier4         -0.27971    0.28715  -0.974  0.33000
## Quartier5         -0.74774    0.32375  -2.310  0.02091 *
## Quartier6         -0.94875    0.51014  -1.860  0.06292 .
## Zuzug2            -0.27088    0.39418  -0.687  0.49197
## Zuzug3            -0.43000    0.42155  -1.020  0.30771
## Zuzug4            -0.86657    0.44615  -1.942  0.05210 .
## Zuzug5            -1.25809    0.47434  -2.652  0.00799 **
## Zuzug6            -1.07209    0.42971  -2.495  0.01260 *
## Zuzug7            -1.68355    0.39581  -4.253 2.11e-05 ***
## Bildung4           0.16950    0.43014   0.394  0.69355
## Bildung7          -0.45277    0.23781  -1.904  0.05692 .
## Bildung8           0.25791    0.35234   0.732  0.46418
## HH2                0.46015    0.43526   1.057  0.29043
## HH3                0.02129    0.31228   0.068  0.94564
## HH4               -0.33615    0.36590  -0.919  0.35826
## HH5                0.02688    0.33711   0.080  0.93644
## Kinder_Schule2     0.52142    0.32813   1.589  0.11204
## Erwerb2           -0.05718    0.23857  -0.240  0.81058
## Erwerb3           -0.16791    0.51440  -0.326  0.74411
## Erwerb4            0.55443    0.47412   1.169  0.24225
```

```
## Erwerb5            -0.12117    0.40622  -0.298  0.76549
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 853.90  on 615  degrees of freedom
## Residual deviance: 742.88  on 582  degrees of freedom
## AIC: 810.88
##
## Number of Fisher Scoring iterations: 13
```

Interpretation

For the knowledge about ways to communicate personal issues, the most significant variable seems to be, how long a person has lived in the city. This makes sense, since people who have been living in the city for a longer period of time are probably better informed about possibilities of participation.

```r
# without Interaction
lm.knw.participate <- glm(knw.participate ~ Geschlecht + Altergruppe + Beziehungstatus +
                            Quartier + Zuzug + Bildung + HH + Kinder_Schule + Erwerb,
               family = "binomial",
               data = df.small)

summary(lm.knw.participate)
```

```
##
## Call:
## glm(formula = knw.participate ~ Geschlecht + Altergruppe + Beziehungstatus +
##     Quartier + Zuzug + Bildung + HH + Kinder_Schule + Erwerb,
##     family = "binomial", data = df.small)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3867  -1.0372   0.6344   0.8842   1.9209
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        1.72171    1.29466   1.330 0.183567
## Geschlecht2        0.39383    0.19401   2.030 0.042363 *
## Geschlecht3      -14.36244  882.74345  -0.016 0.987019
## Altergruppe2       0.62844    1.15653   0.543 0.586866
## Altergruppe3       0.70986    1.15224   0.616 0.537845
## Altergruppe4       0.77498    1.15430   0.671 0.501975
## Altergruppe5       0.29070    1.16256   0.250 0.802549
## Altergruppe6      -0.15358    1.17375  -0.131 0.895899
## Altergruppe7      -0.14913    1.22182  -0.122 0.902853
## Altergruppe8     -14.35071  882.74427  -0.016 0.987029
## Beziehungstatus2  -0.11116    0.27626  -0.402 0.687419
## Quartier2         -0.36211    0.29564  -1.225 0.220646
## Quartier3         -0.08718    0.27724  -0.314 0.753177
## Quartier4         -0.07696    0.29722  -0.259 0.795688
## Quartier5         -0.33435    0.32574  -1.026 0.304689
## Quartier6         -0.84875    0.49065  -1.730 0.083658 .
```

```
## Zuzug2            -0.74880    0.44993  -1.664 0.096061 .
## Zuzug3            -0.68489    0.47993  -1.427 0.153566
## Zuzug4            -0.95689    0.49810  -1.921 0.054722 .
## Zuzug5            -1.29266    0.51619  -2.504 0.012271 *
## Zuzug6            -0.89588    0.47591  -1.882 0.059773 .
## Zuzug7            -1.67776    0.44121  -3.803 0.000143 ***
## Bildung4           0.02162    0.46433   0.047 0.962870
## Bildung7          -0.90144    0.24984  -3.608 0.000309 ***
## Bildung8           0.12090    0.37516   0.322 0.747258
## HH2                0.18351    0.45148   0.406 0.684394
## HH3                0.08918    0.32521   0.274 0.783926
## HH4               -0.18850    0.37054  -0.509 0.610946
## HH5               -0.37690    0.35517  -1.061 0.288598
## Kinder_Schule2     0.26296    0.31813   0.827 0.408478
## Erwerb2           -0.32393    0.23962  -1.352 0.176420
## Erwerb3           -0.79391    0.55226  -1.438 0.150555
## Erwerb4            0.31909    0.48129   0.663 0.507341
## Erwerb5           -0.21033    0.43069  -0.488 0.625301
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 824.57  on 615  degrees of freedom
## Residual deviance: 717.27  on 582  degrees of freedom
## AIC: 785.27
##
## Number of Fisher Scoring iterations: 13
```

Interpretation For the knowledge about possibilities to participate in urban projects, again the *Zuzug* seems to be most significant. Additionally, having an university degree seems to be very important too. This could be the case, because most ot the respondents have an university degree.

```r
# without Interaction
lm.knw.propose <- glm(knw.propose ~ Geschlecht + Altergruppe + Beziehungstatus +
                        Quartier + Zuzug + Bildung + HH + Kinder_Schule + Erwerb,
              family = "binomial",
              data = df.small)

summary(lm.knw.propose)
```

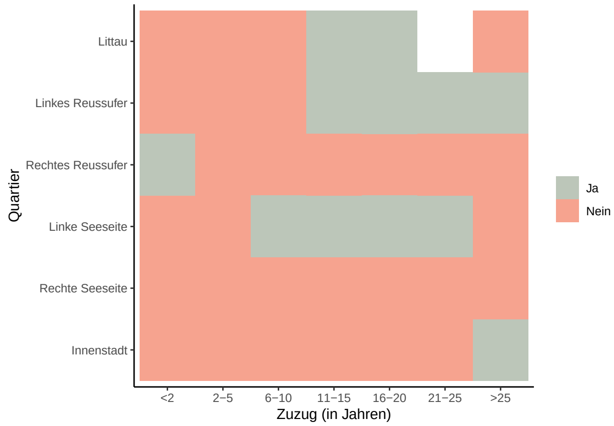
```
##
## Call:
## glm(formula = knw.propose ~ Geschlecht + Altergruppe + Beziehungstatus +
##     Quartier + Zuzug + Bildung + HH + Kinder_Schule + Erwerb,
##     family = "binomial", data = df.small)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5929  -1.0890   0.5921   0.8424   1.5535
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
```

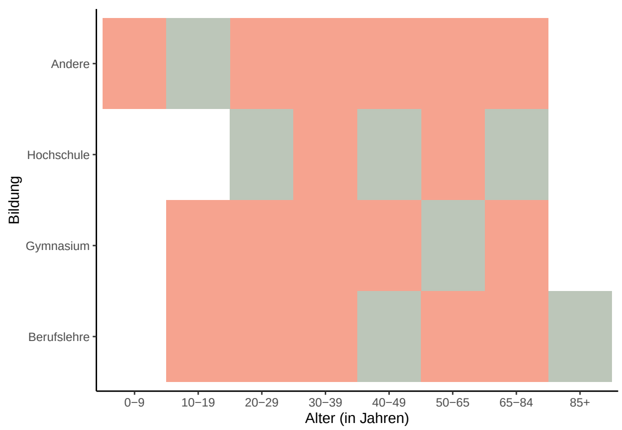
```
## (Intercept)        0.31200    1.32205   0.236  0.81344
## Geschlecht2        0.46614    0.20377   2.288  0.02216 *
## Geschlecht3      -14.63702  882.74345  -0.017  0.98677
## Altergruppe2       1.70799    1.15285   1.482  0.13846
## Altergruppe3       2.41918    1.17100   2.066  0.03884 *
## Altergruppe4       2.42688    1.17071   2.073  0.03817 *
## Altergruppe5       2.35277    1.18354   1.988  0.04682 *
## Altergruppe6       1.73770    1.18964   1.461  0.14410
## Altergruppe7       1.67993    1.22430   1.372  0.17001
## Altergruppe8     -13.15883  882.74431  -0.015  0.98811
## Beziehungstatus2  -0.05519    0.29526  -0.187  0.85173
## Quartier2         -0.56741    0.29810  -1.903  0.05698 .
## Quartier3         -0.10693    0.29294  -0.365  0.71510
## Quartier4         -0.39609    0.30048  -1.318  0.18744
## Quartier5         -0.38889    0.33956  -1.145  0.25209
## Quartier6         -0.73392    0.48523  -1.513  0.13040
## Zuzug2            -0.89861    0.52148  -1.723  0.08485 .
## Zuzug3            -0.41429    0.56479  -0.734  0.46324
## Zuzug4            -1.26400    0.55746  -2.267  0.02336 *
## Zuzug5            -1.39758    0.57460  -2.432  0.01500 *
## Zuzug6            -1.20306    0.54265  -2.217  0.02662 *
## Zuzug7            -1.39285    0.50553  -2.755  0.00587 **
## Bildung4          -0.34997    0.47675  -0.734  0.46291
## Bildung7          -0.58123    0.25694  -2.262  0.02369 *
## Bildung8           0.03337    0.38287   0.087  0.93055
## HH2               -0.17299    0.45591  -0.379  0.70435
## HH3               -0.03338    0.34092  -0.098  0.92201
## HH4               -0.30411    0.38188  -0.796  0.42583
## HH5               -0.04269    0.38416  -0.111  0.91151
## Kinder_Schule2    -0.03788    0.31737  -0.119  0.90499
## Erwerb2           -0.12638    0.24744  -0.511  0.60951
## Erwerb3            0.12375    0.61682   0.201  0.84099
## Erwerb4            0.30391    0.47496   0.640  0.52226
## Erwerb5            0.57207    0.53438   1.071  0.28438
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 761.19  on 615  degrees of freedom
## Residual deviance: 675.78  on 582  degrees of freedom
## AIC: 743.78
##
## Number of Fisher Scoring iterations: 13
```

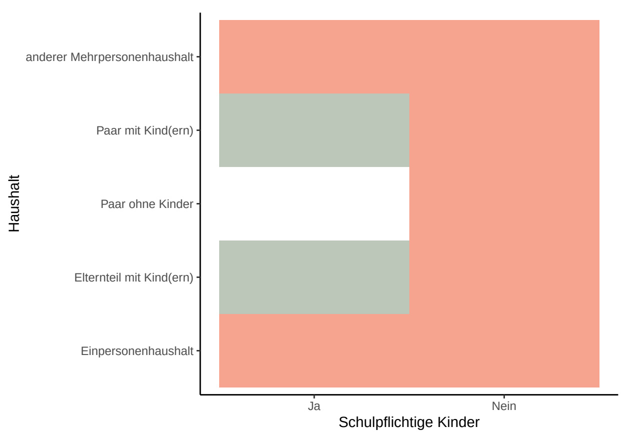
Heatmaps

I further analyse the yes/no questions visually with the healp of heatmaps.

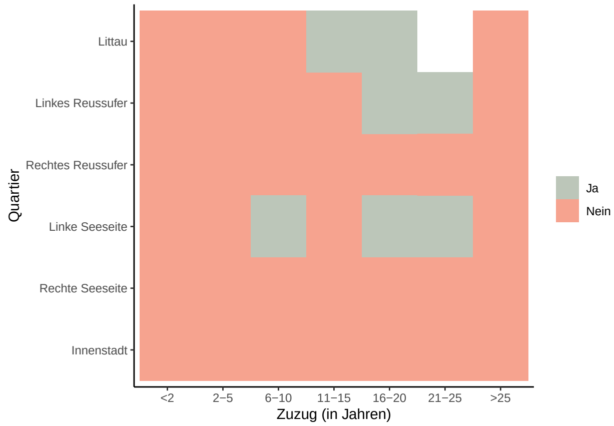
“Kennen Sie Möglichkeiten, um persönliche Anliegen einzubringen?”

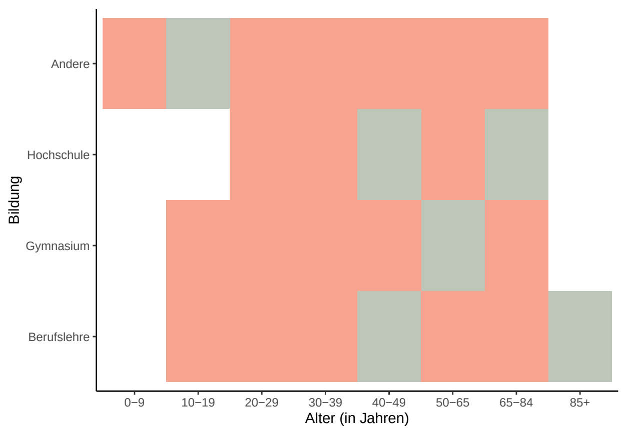


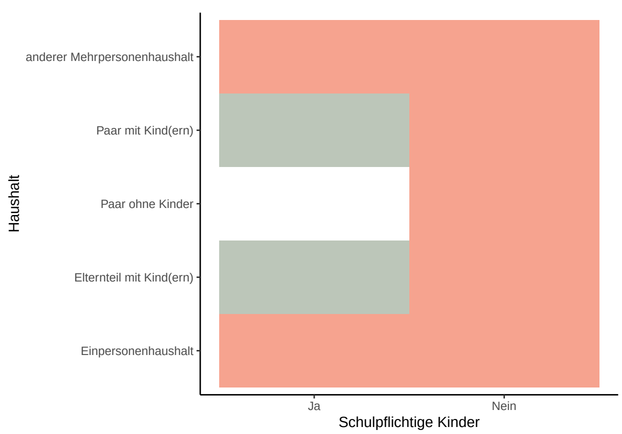




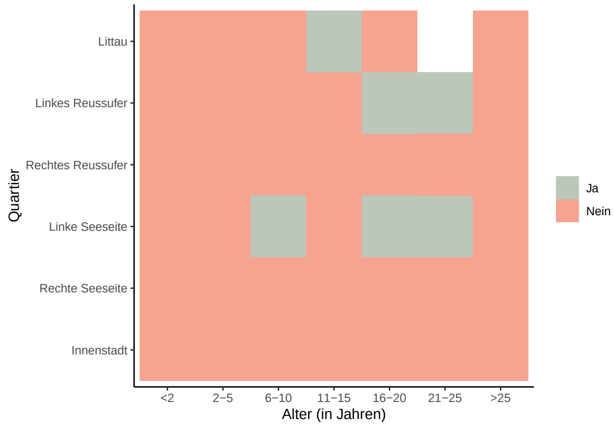
“Kennen Sie Möglichkeiten, sich in städtischen Projekten einzubringen?”

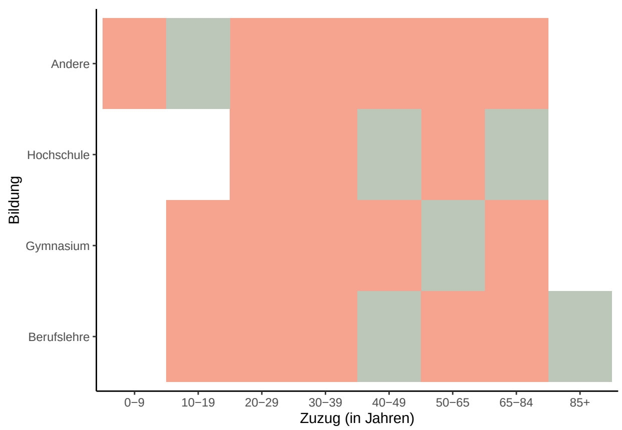


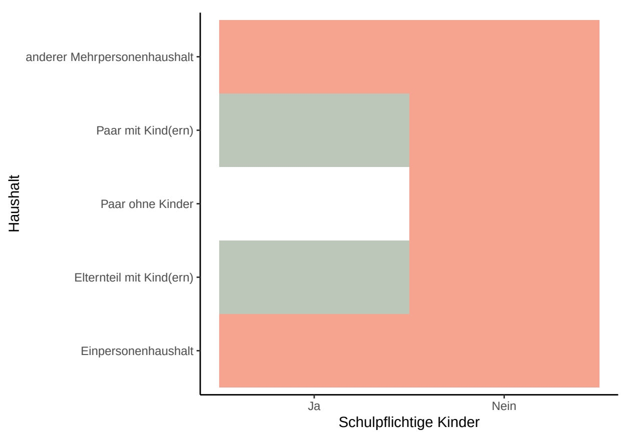




"Kennen Sie Möglichkeiten, städtische Projekte vorzuschlagen?"







Desicion Tree

Desicion Tree for the question *Kennen Sie die Möglichkeiten, sich in städtische Projekte einzubringen?*

```r
# create train data set
set.seed(73)
ratio <- 0.7
total <- nrow(df.small)

train <- sample(1:total, as.integer(total * ratio))
```

```r
table(df.small$knw.participate)
```

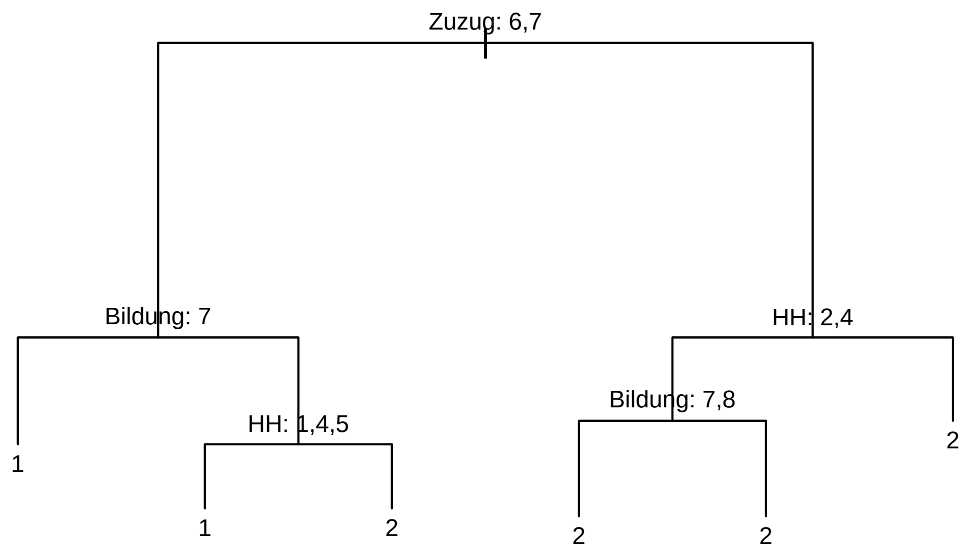
```
##
##   1   2
## 241 375
```

```r
set.seed(1234)
tree.classification.212 <- tree(knw.participate ~ Geschlecht + Altergruppe +
                                  Beziehungstatus + Quartier + Zuzug + Bildung +
                                  HH + Kinder_Schule + Erwerb,
                                data=df.small, subset=train)

summary(tree.classification.212)
```

```
##
## Classification tree:
## tree(formula = knw.participate ~ Geschlecht + Altergruppe + Beziehungstatus +
##     Quartier + Zuzug + Bildung + HH + Kinder_Schule + Erwerb,
##     data = df.small, subset = train)
## Variables actually used in tree construction:
## [1] "Zuzug"   "Bildung" "HH"
## Number of terminal nodes:  6
## Residual mean deviance:  1.192 = 506.4 / 425
## Misclassification error rate: 0.3063 = 132 / 431
```

```r
plot(tree.classification.212)
text(tree.classification.212, pretty=1, cex=0.75)
```



** Interpretation** The model predicts that people who lived less than 16 Years in the city, answer this question with “no”. Based on the model, the others answer “no”, if they don’t have an university degree. For people without a degree, neighborhood and age seem to play a role.

Model Testing

To test the performance of this the model, I train the model with a subset and predict the values for rest of the data.

```
tree.classification.212.pred <- predict(tree.classification.212, df.small[train,], type="class")

# confusion table to determine classification error on *train data*
(tree.classification.212.pred.ct <- table(tree.classification.212.pred,
                                          df.small[train,]$knw.participate))
```

```
##
## tree.classification.212.pred   1   2
##                             1  82  49
##                             2  83 217
```

```
tree.classification.212.pred.correct <- 0
tree.classification.212.pred.error <- 0
for (i1 in 1:2) {
  for (i2 in 1:2) {
    if (i1 == i2) {
```

```r
      tree.classification.212.pred.correct <- tree.classification.212.pred.correct +
        tree.classification.212.pred.ct[i1,i2]
    }else{
     tree.classification.212.pred.error <- tree.classification.212.pred.error +
       tree.classification.212.pred.ct[i1,i2]
    }
  }
}
(tree.classification.212.pred.rate <- tree.classification.212.pred.correct/
    sum(tree.classification.212.pred.ct))
```

```
## [1] 0.6937355
```

```r
# portion of correctly classified observations 51.3%
(tree.classification.212.pred.error <- 1 - tree.classification.212.pred.rate)
```

```
## [1] 0.3062645
```

```r
# train error (pruned): 48.7%

# and on test data --> test error
tree.classification.212.pred.test <- predict(tree.classification.212,
                                             df.small[-train,], type="class")
# confusion table to determine classification error on *test data*
(tree.classification.212.pred.test.ct <- table(tree.classification.212.pred.test,
                                               df.small[-train,]$knw.participate))
```

```
##
## tree.classification.212.pred.test  1  2
##                                 1 33 33
##                                 2 43 76
```

```r
tree.classification.212.pred.correct <- 0
tree.classification.212.pred.error <- 0
for (i1 in 1:2) {
  for (i2 in 1:2) {
    if (i1 == i2) {
      tree.classification.212.pred.correct <- tree.classification.212.pred.correct +
        tree.classification.212.pred.test.ct[i1,i2]
    }else{
      tree.classification.212.pred.error <- tree.classification.212.pred.error +
        tree.classification.212.pred.test.ct[i1,i2]
    }
  }
}
(tree.classification.212.pred.rate <- tree.classification.212.pred.correct/
    sum(tree.classification.212.pred.test.ct))
```

```
## [1] 0.5891892
```

```r
# portion of correctly classified observations 53.0%
(tree.classification.212.pred.error <- 1 - tree.classification.212.pred.rate)
```

```
## [1] 0.4108108
```

```r
# test error (pruned): 47.0%
```

In-sample performance: The model assigns about 70 % right. Out-of-sample Performance: The model assigns about 60%. Therefore, the performance of this model is not very good, as it is only slightley better than a 50/50 chance.